

Remarks

The Applicants believe that this amendment places the subject application in better condition for allowance and in so doing introduces no new issues. Therefore, entry of this Amendment, reconsideration of the application, and allowance of all claims pending herein is respectfully requested.

Claims 1-49 were originally presented in the subject application. By the foregoing amendment, claims 1, 27, 33 and 45-49 have been amended to more particularly point out and distinctly claim the inventive material of the subject invention. Claims 1-49 remain in this case.

Initially, applicants wish to thank the Examiner for the indication of allowability given original claims 10, 16 and 17 if rewritten in independent form.

The Examiner's concerns are addressed separately below in the order raised in the outstanding Office Action.

No new matter has been added.

Rejections under 35 U.S.C. §112

Claims 27-31 and 33-39 stand rejected under 35 U.S.C. §112, second paragraph, as not specifically pointing out and distinctly claiming the subject matter which the applicants regard as their invention.

Applicants has corrected the typographical errors in claims 27 and 33 as identified by the Examiner, replacing (g) with (iii), and therefore respectfully request withdrawal of this ground of rejection.

Rejections under 35 U.S.C. §§102 and 103

Claims 1, 46, 48 and 49, *inter alia*, stand rejected under 35 U.S.C. §102 as anticipated by Smolders (U.S. Patent No. 6,223,338).

The Office action holds independent claims 46, 48 and 49 rejected for the same reasons as claim 1, stating that each is another “version of the claimed method discussed above, claim 1, wherein all claim limitations also have been addressed and/or covered.... Thus, accordingly, this claim is also anticipated by Smolders.”

Claim 45 and 47, *inter alia*, stand rejected under 35 U.S.C. §103 as obvious over Smolders in view of Davidson (U.S. Patent No. 5,664,191). The Office action holds claims 45 and 47 rejected for the same reasons as claim 1, stating that each “is [another] version of the claimed method discussed above, (claims 1 and 20), wherein all claim limitations also have been addressed and/or covered.... Thus, accordingly, this claim is also obvious.”

These rejections, to the extent they might be considered pertinent to the amended claims presented herewith, are respectfully traversed. It is well settled that there is no anticipation unless (1) all the same elements are (2) found in exactly the same situation and (3) are united in the same way to (4) perform the identical function. As amended, applicants' independent claims 1 and 45-49, and the Smolders reference clearly do not have the same elements, or even functional equivalents thereof. To the contrary, there are significant patentable differences between Applicants' recited invention and the method and system disclosed by Smolders.

Each of the independent claims has been provided with a clarifying amendment, to explicitly indicate that the claimed approach involves monitoring data and ‘changing the behavior’ of the run time execution of software code in a target system.

Applicants believe this characteristic was implicit in the subject claims as originally presented, such that no new search is required and no new matter has been added. For example, the originally claim language of ‘replacing [an] instruction with a program flow change instruction’ that directs ‘execution to a buffer’ where it may be monitored, should be viewed as supporting ‘changing the behavior’ of the software code, and monitoring ‘data’. For this reason, the foregoing amendments should be viewed as being supported by the claims as originally filed.

Together, the claim recitations define a structure which is clearly distinct from that disclosed by Smolders, as discussed below.

Claim 1: Searching a Range of Addresses

Claim 1 recites ‘searching a range of addresses within the software code to identify a desired instruction’.

With regard to this claim element, the Office action states that Smolders’ “processor is programmed to generate a trace interrupt after each branch, or at the end of each basic block of code from the currently running program or process.” This citation indicates that Smolders takes a specified action (generates a trace interrupt) at particular points of code execution (i.e., after each branch, or at the end of each basic block of code). However, neither this cited portion, nor any other portion of the Smolders patent, indicates that Smolders effects any discrete ‘searching’ to identify a desired instruction. Rather, Smolders simply executes code in a conventional manner, and then takes a specified action only when, and if, specific instructions are executed, without performing any searching as claimed.

More specifically, instead of searching for desired instructions, Smolders simply makes use of the ‘branch trace enable’ functionality built into the processor (e.g., a PowerPC®). Smolders sets a ‘branch trace enable bit 80’ which causes the processor to automatically ‘generate a trace interrupt after each branch of code’. (Col. 3, lines 53-57). Smolders thus does not search for the instruction in advance, but rather, the hardware is configured to simply wait, and trigger, once it encounters a particular instruction during execution.

Smolders even teaches away from the claimed searching, indicating that his approach advantageously enables branches to be generated ‘without having to know where the basic blocks of code are in advance.’ (Col. 4, lines 3-5).

Accordingly, Smolders should be viewed as not anticipating the claimed invention.

Claim 1: Replacing an Instruction; and Inserting a Routine

Claim 1, as amended, further recites:

‘(b) replacing the desired instruction with a program flow change instruction directing

execution to a buffer, wherein the program flow change instruction is configured to change the behavior relative to that of the desired instruction, of the run time execution of the software code; and (c) inserting a routine into the buffer, the routine having an output instruction and a branch instruction branching to an address of the software code subsequent to the program flow change instruction’.

The Office action cites Smolders, stating that: “[a]t each interrupt, the address of the beginning of the next block is saved which is the address where the interruption came from. Tracing information for the previous block including its address and its size (the current value of a counter) is created. If the current process is a process to be traced, the tracing information is stored in a trace buffer, the counter is reset to zero returning back to the process from the interrupt.”

Neither this citation, nor any other of Smolders’ teachings, anticipate the above-referenced claim elements. After ‘searching a range of addresses... to identify a desired instruction’ as discussed above, the claimed invention replaces the identified instruction ‘with a program flow change instruction... configured to change the behavior relative to that of the desired instruction, of the run time execution’ for tracing. Advantageously, this approach is processor-independent, as it does not rely on any processor-specific functionality, and permits the execution of nominally any portion of code to be effected on the buffer for viewing.

For example, in particular embodiments, the claimed ‘replacing’ includes replacing the identified instruction with a program flow instruction (e.g., a branch instruction) branching to a software scratchpad buffer. In these embodiments, the ‘inserting’ involves placing a user defined routine into the scratchpad buffer. This routine may include code to generate a usable output, code to return execution back to the original code (i.e., back to “the software code subsequent to the program flow change instruction”), and code to retain the proper execution context upon the return. Advantageously, because these embodiments actively replace nominally any identified instruction with a flow change instruction, nominally any portion of the original code may be traced, regardless of what type of processor is being used to execute the code, and whether or not the particular portion of original code includes a branch.

Smolders does not provide this functionality. Instead of actively replacing identified instruction(s) with a ‘program flow change instruction’, Smolders simply utilizes the ‘branch

trace enable' functionality built into the (PowerPC®) processor to automatically 'generate a trace interrupt after each branch of code'. Smolders' discussion of buffering simply refers to how his trace information (as opposed to the executing code itself) may be buffered once an interrupt occurs. Thus, rather than changing the flow of software execution, Smolders simply triggers particular outputs as the code executes in a conventional manner.

An interesting consequence of Smolders' approach, is that although relatively simple in terms of not having to search for particular instructions as discussed above, this approach is effectively unable to trace any code that does not originally generate a branch. In addition, because Smolders' trace interrupts are generated only after each branch in the code, any information regarding the code block immediately preceding the interrupt would be limited to that stored in various registers at the time of the branch, such as address and size as determined by the current value of a counter (which is subsequently re-set to zero). (Col. 2, lines 13-17). As such, it is evident that Smolders does not trace code by executing it where it is visible to a user (e.g., on a bus or buffer) as taught by the instant application.

For each of the foregoing alternative reasons, all the elements of the subject invention are not found in exactly the same situation in Smolders, nor are they united to perform the identical function. Accordingly, applicants submit that Smolders does not anticipate the subject invention as set forth in claim 1. Moreover, since each independent claim 45-49 includes the characteristics of claim 1 as discussed above, claims 45-49 are similarly believed to be unanticipated by the art of record.

Obviousness

Moreover, claims 1 and 45-49 are believed to set forth non-obvious subject matter over Smolders, viewed independently or combined with any other document of record.

As discussed above, Smolders actually teaches away from the concept of searching for a desired instruction discretely from the execution thereof.

In addition, in light of the distinctions discussed above, even assuming *arguendo* that a combination of Smolders with Davidson, et al. was proper, the synthesis of structure would not come within the scope of the invention as claimed.

CONCLUSION

Applicants therefore respectfully request reconsideration and allowance of amended claims 1 and 45-49 presented herewith. The dependent claims are believed allowable for the same reasons as the independent claims from which they depend, as well as for their own additional limitations.

This application is now believed to be in condition for allowance, and such action at an early date is respectfully requested.

Respectfully submitted,



Richard L. Sampson
Attorney for Applicants
Registration No. 37,231

Dated: August 30, 2004

SAMPSON & ASSOCIATES, P.C.
50 Congress Street
Suite 519
Boston, MA 02109
Telephone: (617) 557-2900
Facsimile: (617) 557-0077